

**Difference with cited prior art**

The present invention as defined in claims 1, 7 and 11 utilizes two different threads created in a memory structure, each representing an independent flow of control managed by a program structure. The first thread has two states: 1) a first state processing work for the program structure and 2) a second state undispatched awaiting work to process. The second thread prepares work for the first thread to process, and places the work in a queue. If the first thread is awaiting work, it processes the work in the queue. Otherwise, the first thread then completes any other work and then accesses and processes the work in the thread. After the first thread completes a desired amount of work, it is destroyed.

The Sievert '729 patent describes a different manner of using threads and queues to execute operations. Unlike the present invention, Sievert does not use a "second thread" to place work in a queue for a first thread to execute. Instead, Sievert requires that an application itself, and not a thread, start the thread that is to perform the operation. Column 2, lines 1-7. Moreover, the application must determine if a thread can accept a queued operation. Column 4, lines 30-34. The application cannot simply place the operation in the queue for it to be executed by the thread – instead, the application can only place the operation in the thread while it is in the "EXECUTING state." Column 6, lines 55-57. An operation cannot be placed in the queue if the thread is in the "STOPPING state." Column 8, lines 63-66. Thus, unlike applicant's invention, Sievert's operation cannot be placed in the queue if the thread is in the idle state; Sievert's application is required to determine if the thread can accept an operation. By contrast, applicants' invention permits a "second thread" to place the work in the queue without

condition. Finally, while Sievert's background discusses prior art methods in which threads are destroyed, there is no disclosure or suggestion that the new method disclosed in the '376 patent does so.

Accordingly, applicants' invention is not disclosed or suggested by the Sievert '376 patent.

**Prior invention by applicants**

Notwithstanding the differences between the Sievert '376 patent and the instant invention, applicants are enclosing herewith a Second Declaration under Rule 131 of Harry J. Beatty and Peter Elmendorf, the inventors, and a Declaration under Rule 131 of Jay Anderson, the assignee's in-house patent counsel, that establish facts showing conception and actual reduction to practice of this invention in this country prior to the September 29, 1999 filing date of the Sievert '376 patent cited against this application, and due diligence from a time prior to that date until the instant application was filed and constructively reduced to practice on June 20, 2000.

The declarations establish that each and every limitation of claim 1 of the above-referenced application is disclosed in the attached invention disclosure attached as Exhibit A, created and dated prior to September 29, 1999, but redacted to protect confidential information, and in the drawing attached as Exhibit B. A comparison of the limitations of claim 1 and the Disclosure is as follows:

The invention claimed in the subject application is directed to a method of parallel processing in a memory structure employing first and second threads, with the first thread waiting for and processing work prepared for it by the second thread. The Exhibit A

disclosure uses the term "first thread" in the same manner as used in the claims of the subject application, while the Exhibit B drawing refers to the first thread as "thread (in a bottle)", using the same "bottle" terminology for the first thread as used in column 12, lines 24-26 and Fig. 11 of the subject application. However, the Exhibit A disclosure and Exhibit B drawing use different terms for the "second thread" referenced in the claims; the Exhibit A disclosure uses the term "software" and the Exhibit B disclosure uses the term "escapement" for what is described in the claims as the "second thread," and which is also described in the subject application as the "launcher." See column 12, lines 52-56 and Fig. 11. Although the concept is the same, in preparing the subject application the inventors decided to use the term "launcher" instead of the term "escapement" since the former would be more widely understood.<sup>1</sup> Declaration, ¶5. Thus, the Exhibit B drawing shows the actions of the claimed first thread in the left side flow chart labeled "Create Thread (in a bottle)" and the actions of the claimed second thread in the center flow chart labeled "escapement (work, fcn)."

The following chart compares the steps of the claimed method in detail with the teachings in the original disclosure and drawing of Exhibits A and B:

---

<sup>1</sup> The term "escapement" is defined as "a device in a timepiece which controls the motion of the train of wheelwork and through which the energy of the power source is delivered to the pendulum or balance by means of impulses that permit a tooth to escape from a pallet at regular intervals." Merriam-Webster OnLine Dictionary, [www.m-w.com](http://www.m-w.com). Thus, an "escapement" is analogous to the "launcher" or "second thread" which controls the pace of the first thread by assigning it work.